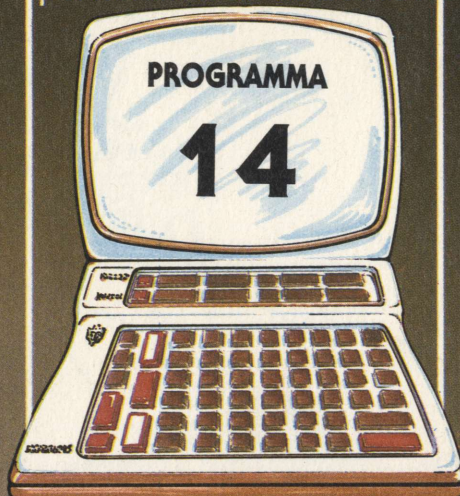
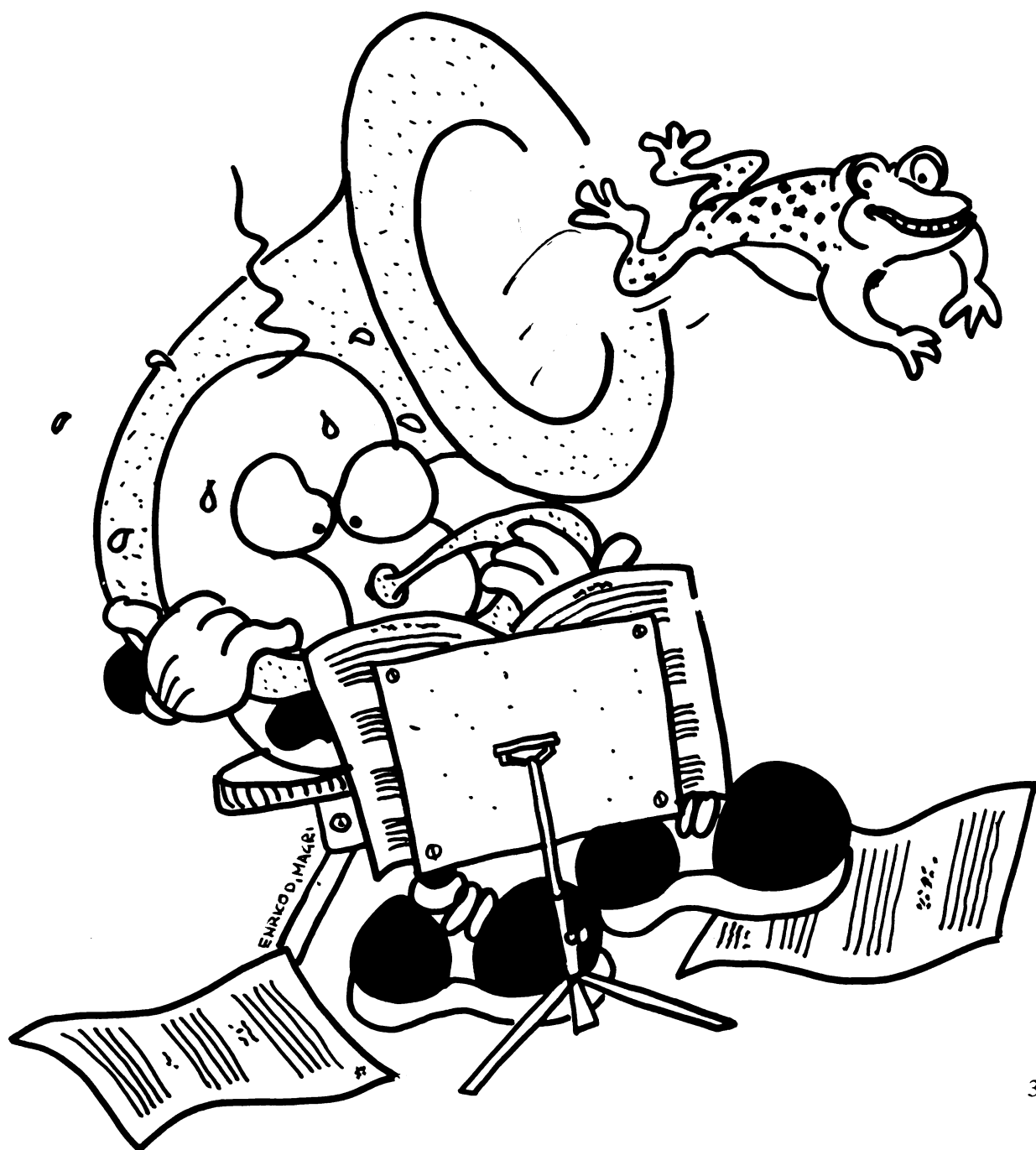
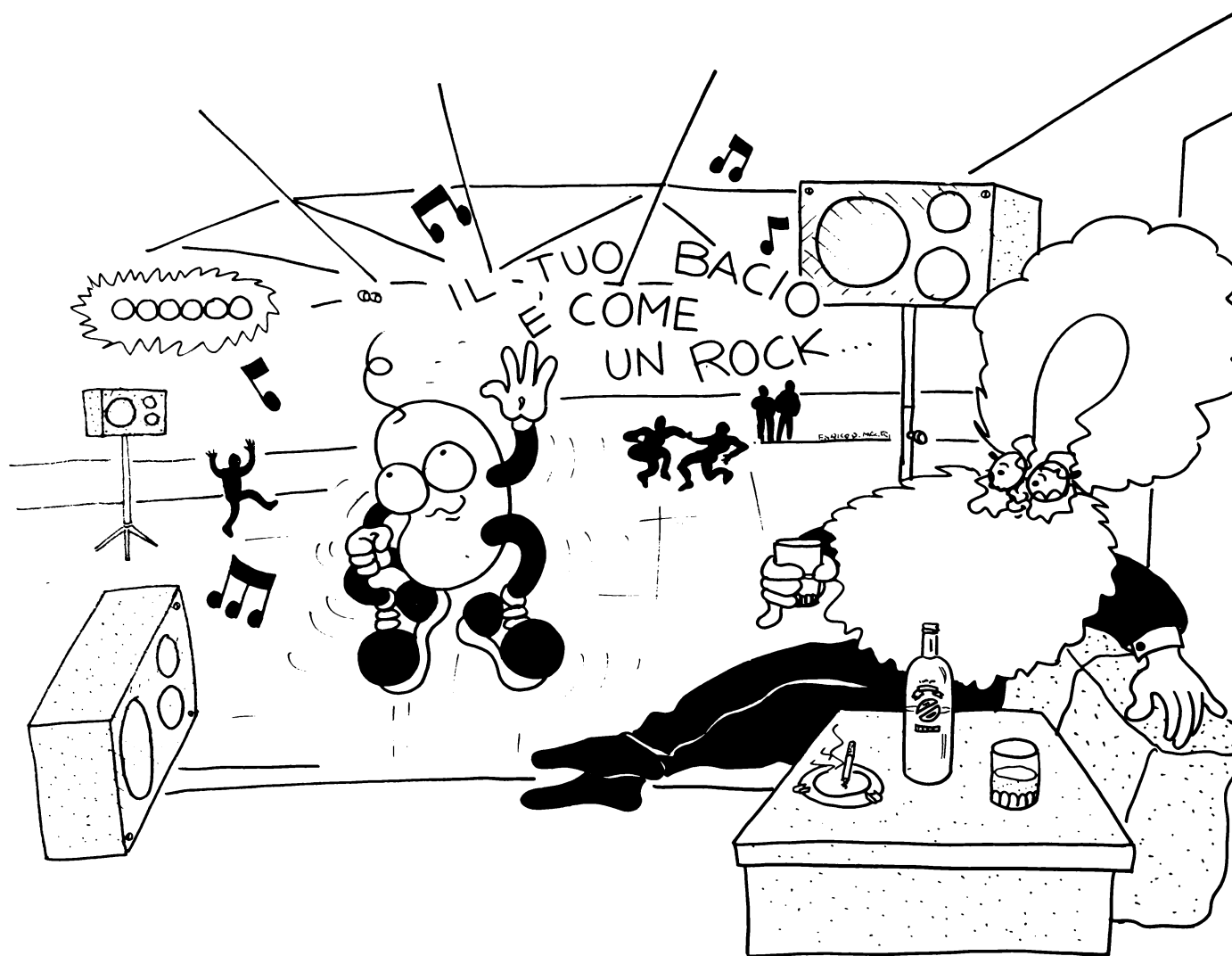
# CONOSCERE IL COMPUTER DIRETTAMENTE DAL COMPUTER

## per Commodore Vic20 e 64

Beatrice d'Este

Nell'allegato alla lezione 25, **IL SUONO**, abbiamo visto le locazioni di memoria su cui agire per creare effetti sonori e musicali, cercando in questo modo di capire la struttura di un programma sonoro attraverso la spiegazione e l'uso delle funzioni principali: volume, forme d'onda, ADSR, frequenza, filtri, ecc.

Ma, oltre che per la costruzione di un programma sonoro, i suoni possono essere utilizzati anche in molti altri modi, e questo è appunto l'argomento della lezione 27.

Infatti i suoni, non solo possono essere usati nei video giochi (spari, esplosioni, rumori, ecc.) ma anche, come vedremo, in programmi gestionali o educativi; quando c'è bisogno di sottolineare, ad esempio, alcune domande o risposte con un "beep", oppure di associare certi colori a determinati suoni.

Nella costruzione di un programa musicale solitamente conviene usare le parti del programma che creano i suoni come subroutine e, per evitare di dover poi ricopiare nei poke le locazioni di memoria a cui accedere, è preferibile memorizzare tutti i numeri delle locazioni che regolano i generatori di suoni in opportune variabili.

# Listato dell'esercizio: ESEMPIO PRATICO (CBM 64)

```
10 si=54272
20 pokesi+24,15
30 pokesi+4,33
40 pokesi+5,15:pokesi+6,0
50 forf=1to150
60 pokesi,0:pokesi+1,f
70 fork=1to100:next
80 nextf
90 pokesi+4,0
100 pokesi+24,0
```

# Listato dell'esercizio: ESEMPIO PRATICO (VIC 20)

```
10 poke36878,15
20 forf=128to250
30 poke36876,f
40 nextf
50 poke36876,0
```

# Listato dell'esercizio: COMPLETA IL LISTATO (CBM 64)

```
10 poke53280,6:poke53281,3
20 n=0:printchr$([*])
30 l1=24:l2=38:x=0:[*]
40 forg=[*]to6
50 l1=l1-2:for[*]tox+l1
60 gosub150:nextx
70 l2=[*]-2:fory=ytoy+l2
80 gosub150:next[*]
90 if[*]thenl1=l1+1
100 l1=[*]-2:forx=xtox-l1step-1
110 gosub150:nextx
120 l2=l2-[*]:fory=ytoy-l2step-1
130 gosub[*]:nexty
140 nextg:[*]
150 p=x*40+y:poke55296+p,2
160 poke1024+p,160:[*]
```

# Listato dell'esercizio: COMPLETA IL LISTATO (VIC 20)

```
10 poke36879,62
20 n=0:printchr$([*])
30 l1=22:l2=20:x=0:[*]
40 forg=[*]to5
50 l1=l1-2:for[*]tox+l1
60 gosub150:nextx
70 l2=[*]-2:fory=ytoy+l2
80 gosub150:next[*]
90 if[*]thenl1=l1+1
100 l1=[*]-2:forx=xtox-l1step-1
110 gosub150:nextx
120 l2=l2-[*]:fory=ytoy-l2step-1
130 gosub[*]:nexty
140 nextg:[*]
150 p=x*22+y:poke38400+p,2
160 poke7680+p,160:[*]
```

# PROGRAMMIAMO INSIEME (CBM 64)

```
10 poke53280,3:poke53281,3:poke650,128
20 print"■"chr$(142)
30 v=1024:c=55296:q=40:p=940
40 a=65:n=83:m=46:ca=0:cn=6:cs=3:cm=2
100 l=p:getx$:gosub2000
110 ifx$=""then100
120 if (x$="z")*(p>920) then p=p-1
130 if (x$="x")*(p<959) then p=p+1
140 if x$=" " then gosub 1000
200 pokec+l,cs:pokev+l,32
220 pokec+p,ca:pokev+p,a
300 goto100
1000 h=p-q:f=h:gosub3000
1040 pokec+f,cs:pokev+f,32
1060 pokec+h,cm:pokev+h,m
1070 if h=g then gosub 3000:f=h:g=900:pn=pn+1:
print"■punteggio";pn:goto 1100
1080 f=h:h=h-q:if h>q then1040
1100 pokec+f,cs:pokev+f,32:return
2000 t=g:if g>840 then j=j+1:g=int(rnd(0)*q)
```

```
2010 if j=20 then end
2020 g=g+q
2040 pokec+t,cs:pokev+t,32
2060 pokec+g,cn:pokev+g,n
2100 return
3000 si=54272
3020 for y=15 to 0 step -1
3040 pokesi+24,y:pokesi+5,15:pokesi+6,0:pokesi+1,40:
pokesi,200:pokesi+4,129
3060 next y:pokesi+4,0:pokesi+5,0:return
```

# PROGRAMMIAMO INSIEME (VIC 20)

```
10 poke36879,59:poke650,128
20 print"■"chr$(142)
30 v=7680:c=38400:q=22:p=473
40 a=65:n=83:m=46:ca=0:cn=6:cs=3:cm=2
100 l=p:getx$:gosub2000
110 ifx$=""then100
120 if (x$="z")*(p>462) then p=p-1
130 if (x$="x")*(p<483) then p=p+1
140 if x$=" " then gosub 1000
200 pokec+l,cs:pokev+l,32
220 pokec+p,ca:pokev+p,a
300 goto100
1000 h=p-q:f=h:gosub3000
1040 pokec+f,cs:pokev+f,32
1060 pokec+h,cm:pokev+h,m
1070 if h=g then gosub 3000:f=h:g=550:pn=pn+1:print
"■punteggio";pn:goto 1100
1080 f=h:h=h-q:if h>q then1040
1100 pokec+f,cs:pokev+f,32:return
2000 t=g:if g>418 then j=j+1:g=int(rnd(0)*q)
2010 if j=20 then end
2020 g=g+q
2040 pokec+t,cs:pokev+t,32
2060 pokec+g,cn:pokev+g,n
2100 return
3000 poke36878,15
3020 for y=200 to 128 step -1
3040 poke36876,y
3060 next y:poke36876,0:poke36878,0:return
```

# Soluzione dell'esercizio: COMPLETA IL LISTATO CBM 64
(lez. 26)

```
10 poke53280,6:poke53281,3:c=54272
20 printchr$(142)chr$(147)
30 forx=40to999
40 ifint(rnd(0)*50)<46then60
50 poke1024+x,86:poke55296+x,6
60 nextx:k=1064:a=ti
70 geta$:pokek,88:pokec+k,1:pk=k
80 ifa$="x"then k=k+1
90 ifa$="z"then k=k-1
100 ifa$=";"then k=k-40
110 ifa$="/"then k=k+40
120 ifpk<>kthenpokec+pk,3
130 if(k<1064)+(k>2023)thenk=1064
140 ifpeek(k)=86thens=s+1:pokec+k,3
150 printchr$(19)tab(15)"punti:";s
160 ifti-a<1800then70
```

# Soluzione dell'esercizio: COMPLETA IL LISTATO VIC 20
(lez. 26)

```
10 poke36879,62:c=30720
20 printchr$(142)chr$(147)
30 forx=22to505
40 ifint(rnd(0)*50)<46then60
50 poke7680+x,86:poke38400+x,6
60 nextx:k=7702:a=ti
70 geta$:pokek,88:pokec+k,1:pk=k
80 ifa$="x"then k=k+1
90 ifa$="z"then k=k-1
100 ifa$=";"then k=k-22
110 ifa$="/"then k=k+22
120 ifpk<>kthenpokec+pk,3
130 if(k<7702)+(k>8185)thenk=7702
140 ifpeek(k)=86thens=s+1:pokec+k,3
150 printchr$(19)tab(6)"punti:";s
160 ifti-a<1800then70
```